Telescr.in

Audit of smart contract SmartX

revision 2 dated 15.08.2024

Telescr.in

Summary

Brief information

Project: SmartX
Network: BNB
Compiler version: 0.8.26
Optimization: Enabled
Audit date: 15.08.2024

Information

The contract code was reviewed and analysed for vulnerabilities, logical errors and developer exit scams possibilities. This work was carried out concerning the project source code and documentation provided by the customer.

Customer provided project documentation, the audit was made based on provided document.

General conclusion

As a result of the audit, no errors were found that affect the security of users' funds on the contract. No obvious signs of an exit scam were found. No backdoors found either. The contract is highly dependent on owner's behaviour.

Liability disclaimer

The telescr.in team within this audit framework is not responsible for the developers or third parties' actions on the platforms associated with this project (websites, mobile applications, and so on). The audit confirms and guarantees only the smart contract correct functioning in the revision provided by the project developers.

Confirmed by digital signature

Aggregated data

The Contract analysis was performed using the following methods:

- Static analysis
  - Checking the code for common errors leading to the most common vulnerabilities
- Dynamic analysis
  - The Contract launching and carrying out the attacks various kinds to identify vulnerabilities
- Code Review

Received data

Recommendation	Type	Priority	Occurrence probability	Line of error
Useless Definition	remark	low		11-14
Some public functions might be external	notice	low		1
Strong dependency on the contract's owner actions	warning	medium	low	1

A. Errors

Not found.

B. Warnings

1. Strong dependency on the contract's owner actions

The contract is highly dependent on owner's behaviour. E.g. in the function setAccount the owner is allowed to set accounts; also, they can set any incomeAmount and depositedValue to any accountAddress. The only restriction is the check whether the contract is sealed, but the function sealContract is marked as onlyOwner as well.

During the re-audit process, the contract was sealed at the transaction 0x01d3504f560daae3a11469202d4069cd5641900326149c1e0488b939e56394c7. Thus, it isn't possible to change and/or set an account by the owner anymore

C. Notice

1. Some public functions might be external

Some setter functions as setAccount, sealContract, setMarketingAddress, setFeeAddress, etc... are never called within the contract. Thus, it can be defined as external in porpose to save gas. [status: fixed]

D. Remarks

1. Useless Definition

enum Side is defined, but never used properly. Type of the enum is only passed to the function register.

Developer team's comment: It was added for integration with the binary referral system purpose

Application. Error classification

Priority	
informational	This question is not directly related to functionality but may be important to understand.
low	This question has nothing to do with security, but it can affect some behavior in unexpected ways.
medium	The problem affects some functionality but does not result in an economically significant user funds loss.
high	This issue can result in the user funds loss.
Probability	
low	It is unlikely that the system is in a state in which an error could occur or could be caused by any party.
medium	This problem may likely arise or be caused by some party.
high	It is highly likely that this problem could arise or could be exploited by some parties.

Application. Digital bytecode print

The audit was carried out for the code certain version on the compiler version 0.8.26 with the optimization enabled.

To check the contract bytecode for identity to the one that was analyzed during the audit, you must:

. Get contract bytecode (in any block explorer)
. Get SHA1 from bytecode string
. Compare with reference in this report

Sha1 from bytecode:
22abb47886a88c88cce89db83b32c673b6a817f6
Sha1 from bytecode (non-metadata):
2f997851599b94c76391610b89dcadd4b3989cb4
Contract address:
0x5EC643C00d4F62B45C248920991188367A1FaEB0

Check the digital print

Application. Signature of the audit report

{ "address": "0x505ade8cea4db608250e503a5e8d4cb436044d2e", "msg": "As a result of the audit, no errors were found that affect the security of users' funds on the contract. No obvious signs of an exit scam were found. No backdoors found either. The contract is highly dependent on owner's behaviour. . Sha1 of contract - 22abb47886a88c88cce89db83b32c673b6a817f6. Sha1 without meta of contract - 2f997851599b94c76391610b89dcadd4b3989cb4Contract address - 0x5EC643C00d4F62B45C248920991188367A1FaEB0", "sig": "0x79cf03b7f9972b39b2ee5bcc5e6fdf306eef837bfaee3664d7c8930cdaaf9185692b407cf81f976a4fa5e83311cd9a47d3ead8f74e66f17d8f102d150a0f86c21b", "version": "3", "signer": "MEW" }

Check the signature